Estimation de mouvement dense entre images distantes : intégration combinatoire multi-steps et sélection statistique

Pierre-Henri CONZE[1,2], Tomás CRIVELLI[1], Philippe ROBERT[1], Luce MORIN[2]

[1]Technicolor
975 avenue des Champs Blancs CS 17616, 35576 Cesson-Sévigné Cedex, France

[2]INSA Rennes, IETR/UMR 6164, UEB
20 avenue des Buttes de Coesmes, 35708 Rennes Cedex 7, France

pierre-henri.conze@technicolor.com, tomas.crivelli@technicolor.com,
philippe.robert@technicolor.com, luce.morin@insa-rennes.fr

Résumé – Pour traiter le problème de la mise en correspondance dense entre images distantes, nous proposons une méthode d'intégration combinatoire *multi-steps* permettant de construire un grand ensemble de champs de mouvement candidats via de multiples chemins de mouvement. Une sélection du champ optimal est ensuite réalisée en utilisant, en plus des techniques d'optimisation globale couramment utilisées, un traitement statistique exploitant la densité spatiale des candidats ainsi que leur cohérence *forward-backward*. Les expériences réalisées dans le domaine de l'édition vidéo montrent les bonnes performances que notre méthode permet d'obtenir.

Abstract – To address the problem of dense motion estimation between distant frames, we present a combinatorial *multi-step* integration procedure which allows one to obtain a large set of candidate motion fields between the two distant frames by considering multiple motion paths across the video sequence. Given this large set, we propose to perform the optimal motion vector selection by combining a global optimization stage with a new statistical processing which exploits the spatial distribution of candidates and introduces an intra-candidate quality based on *forward-backward* consistency. Experiments show the effectiveness of our method for distant motion estimation in the context of video editing.

1 Introduction

Bien que robustes entre images consécutives, les méthodes d'estimation de mouvement de l'état de l'art [1, 2, 3, 4, 5] fonctionnent généralement moins bien entre images distantes. Une mise en correspondance directe entre images non-consécutives peut s'avérer incorrecte, notamment dans les situations complexes suivantes : variations d'illumination, occultations temporaires, zoom, déformations non-rigides, transparence... Une alternative consiste à accumuler des vecteurs de flots optiques élémentaires entre les deux images considérées. Ces vecteurs élémentaires peuvent être calculés entre images consécutives ou davantage éloignées. Dans [6, 7], une méthode d'estimation séquentielle repose sur la fusion à chaque image de cartes de mouvement reliant cette image à la première. Chaque carte candidate est obtenue en concaténant un champ estimé à une image précédente et un champ élémentaire issu d'un estimateur standard. Des champs élémentaires sont donc estimés avec des pas variables (*multi-steps*) entre paires d'images. Les critères généralement considérés pour l'estimation du flot optique ne sont pas adaptés à la sélection de vecteurs de mouvement entre images distantes à cause des hypothèses restrictives dont ils découlent. C'est pourquoi nous suggérons de baser cette sélection sur un critère statistique exploitant la distribution spatiale d'un large ensemble de candidats ainsi que leur qualité intrinsèque.

Nous proposons ainsi une méthode d'intégration combinatoire consistant à construire un grand ensemble de champs de mouvement candidats issus de multiples concaténations *multi-steps* entre deux images distantes. Une sélection du meilleur champ de mouvement parmi les candidats générés est réalisée en combinant traitement statistique et optimisation globale. Nous présentons des résultats d'estimation de mouvement dense entre images distantes dans le domaine de l'édition vidéo.

2 Construction de vecteurs de mouvement candidats

Considérons une séquence de $N+1$ images RGB $\{I_n\}$ avec $n \in [\![0, \dots, N]\!]$. Soient I_a et I_b deux images distantes ($0 \leq a < b \leq N$) entre lesquelles nous souhaitons estimer le mouvement. Définissons un chemin de mouvement comme une concaténation de flots optiques élémentaires le long de la séquence. Soit $S_n = \{s_1, s_2, \dots, s_{Q_n}\} \subset \{1, \dots, N-n\}$ l'ensemble des Q_n *steps* disponibles à l'instant n. Cela signifie que les flots optiques élémentaires $\{\boldsymbol{v}_{n,n+s_1}, \boldsymbol{v}_{n,n+s_2}, \dots, \boldsymbol{v}_{n,n+s_{Q_n}}\}$ peuvent ainsi être utilisés à partir de l'image I_n.

La méthode proposée consiste à générer tout d'abord toutes les séquences de *steps* possibles entre I_a et I_b. Chacune d'entre elles définit, après concaténation des flots optiques correspondants, un chemin de mouvement reliant chaque pixel $\boldsymbol{x}_a$ dans I_a à une position sub-pixélique dans I_b.

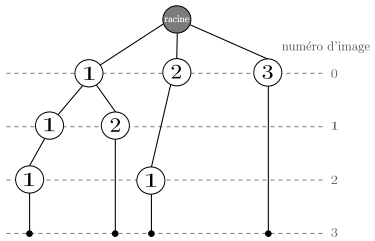


FIGURE 1 – Génération de $\mathbf{\Gamma}_{a,b}$, l'ensemble des séquences de *steps* possibles entre I_a et I_b.

Définissons $\mathbf{\Gamma}_{a,b}$ comme étant l'ensemble des K séquences de *steps* $\boldsymbol{\gamma}_i$ possibles entre I_a et $I_b : \mathbf{\Gamma}_{a,b} = \{\boldsymbol{\gamma}_0, \boldsymbol{\gamma}_1, \ldots, \boldsymbol{\gamma}_{K-1}\}$. $\mathbf{\Gamma}_{a,b}$ est calculé en construisant un arbre (Fig. 1) pour lequel chaque nœud correspond à un champ de mouvement défini pour une image donnée et pour une valeur de *step* donnée (valeur du nœud). L'arbre est crée récursivement à partir du nœud racine en générant autant de nœuds fils que de *steps* disponibles à l'instant courant. La génération d'une branche prend fin lorsque I_b est atteint ou dépassé. Les séquences de *steps* sont obtenues en parcourant l'arbre du nœud racine aux feuilles. Ainsi, l'arbre présenté Fig. 1 indique que quatre séquences de *steps* peuvent être générées entre I_0 et I_3 avec les *steps* 1, 2 et $3 : \mathbf{\Gamma}_{0,3} = \{\{1,1,1\},\{1,2\},\{2,1\},\{3\}\}$.

Une fois toutes les séquences de *steps* $\boldsymbol{\gamma}_i \in \mathbf{\Gamma}_{a,b}$ obtenues, la génération des chemins de mouvement est ensuite effectuée par intégration via la méthode d'*Euler* (Fig. 2). En partant de chaque pixel $\boldsymbol{x}_a$ de I_a et pour chaque séquence de *steps* $\boldsymbol{\gamma}_i$ $\forall i \in [\![0, \ldots, K-1]\!]$, l'intégration consiste à accumuler les flots optiques élémentaires dont les *steps* correspondent à ceux constituant la séquence de *steps* courante. Ainsi, Fig. 2 illustre la construction des quatre chemins de mouvement possibles (un pour chaque séquence de *steps* de $\Gamma_{0,3}$) entre I_0 et I_3 avec les *steps* 1, 2 et 3. Soit $f_j^i = a + \sum_{k=0}^{j} s_k^i$ le numéro d'image courant durant la construction du chemin de mouvement i à partir de I_a où j correspond à l'index de *step* au sein de la séquence de *steps* $\boldsymbol{\gamma}_i$. Pour chaque $\boldsymbol{\gamma}_i \in \mathbf{\Gamma}_{a,b}$ et pour chaque *step* $s_j^i \in \boldsymbol{\gamma}_i$, l'intégration débute en partant de x_a et se poursuit itérativement comme suit :

$$\boldsymbol{x}_{f_j^i} = \boldsymbol{x}_{f_{j-1}^i} + \boldsymbol{v}_{f_{j-1}^i, f_j^i}(\boldsymbol{x}_{f_{j-1}^i}) \tag{1}$$

En parcourant tous les *steps* $s_j^i \in \boldsymbol{\gamma}_i$, nous obtenons $\boldsymbol{x}_b^i$, la position correspondante à $\boldsymbol{x}_a$ dans I_b via γ_i. L'ensemble des séquences de *steps* permettent d'obtenir l'ensemble des positions candidates dans $I_b : T_{a,b}(\boldsymbol{x}_a) = \{\boldsymbol{x}_b^i\}$ avec $i \in [\![0, \ldots, K_{\boldsymbol{x}_a} - 1]\!]$ où $K_{\boldsymbol{x}_a}$ est le cardinal de $T_{a,b}(\boldsymbol{x}_a)$. Des informations d'occultation, fournies avec les flots optiques élémentaires d'entrée, peuvent être utilisées pour arrêter la construction d'un chemin de mouvement si l'un des flots optiques est considéré comme étant occulté. Ainsi, la position $\boldsymbol{x}_{f_j^i}$ ne peut être créée que si le vecteur de flot optique $\boldsymbol{v}_{f_{j-1}^i, f_j^i}$ partant de la position pixel la plus proche de $\boldsymbol{x}_{f_{j-1}^i}$ est non-occulté pour ce *step*.

En pratique, les séquences de *steps* ne peuvent pas être toutes prises en compte en raison de problèmes mémoires et calculatoires. 5877241 chemins de mouvement peuvent être générés pour une distance de 30 images avec les *steps* 1, 2, 5 et 10 par exemple. C'est pourquoi la procédure décrite ci-dessus est restreinte à un sous-ensemble des chemins de mouvement. Ceux dépassant un nombre maximum de concaténations N_c sont supprimés. De plus, une sélection aléatoire de N_s chemins de mouvement parmi ceux restants est réalisée. Celle-ci est guidée par le principe selon lequel la fréquence d'apparition d'un *step* donné à un instant donné doit être uniforme par rapport aux autres *steps* disponibles. Cela évite tout biais envers les branches de l'arbre les plus peuplées.

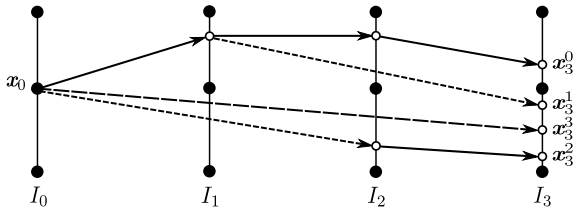


FIGURE 2 – Génération des chemins de mouvement *multi-steps* reliant chaque pixel $\boldsymbol{x}_a$ de I_a à un ensemble de positions candidates dans I_b.

Nous proposons de combiner des chemins de mouvement *forward* et *backward* lors de la construction des candidats associés à $\boldsymbol{x}_a$ dans I_b. Similairement au calcul de $T_{a,b}(\boldsymbol{x}_a)$, de multiples positions candidates dans I_a associées aux pixels $\boldsymbol{x}_b$ de I_b peuvent être calculées via des chemins de mouvement *backward*. Ces derniers peuvent être inversés pour obtenir de nouveaux chemins de mouvement *forward* et peuvent ainsi être utilisés pour enrichir $T_{a,b}(\boldsymbol{x}_a)$ de nouvelles positions candidates. Ces candidats, issus de chemins de mouvement *backward* inversés, sont définis comme des candidats *inverses*. Dans le cas contraire, les candidats sont qualifiés de *directs*.

3 Sélection de vecteurs de mouvement optimaux

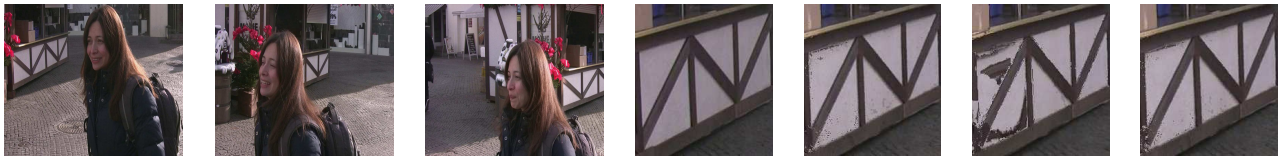
Nous souhaitons désormais sélectionner pour chaque pixel x_a de I_a la position candidate optimale $\boldsymbol{x}_b^*$ parmi l'ensemble des positions candidates dans I_b, $T_{a,b}(\boldsymbol{x}_a) = \{\boldsymbol{x}_b^i\}$ où $i \in [\![0, \ldots, K_{\boldsymbol{x}_a} - 1]\!]$, obtenu lors de l'étape de construction des vecteurs de mouvement candidats. Avec l'hypothèse d'un modèle *Gaussien* décrivant la distribution spatiale de $T_{a,b}(\boldsymbol{x}_a)$, cette étape de sélection se résume à obtenir la valeur centrale de la distribution. Nous utilisons pour cela l'estimateur du maximum de vraisemblance :

$$\boldsymbol{x}_b^* = arg\min_{\boldsymbol{x}_b^i} \sum_{\substack{j=0 \\ j \neq i}}^{K_{\boldsymbol{x}_a}-1} \left\| \boldsymbol{x}_b^j - \boldsymbol{x}_b^i \right\|_2^2 \tag{2}$$

En remplaçant la moyenne par la médiane pour être plus robuste aux valeurs aberrantes, le choix de $\boldsymbol{x}_b^*$ est définit par :

$$\boldsymbol{x}_b^* = arg\min_{\boldsymbol{x}_b^i} med_{j \neq i} \left\| \boldsymbol{x}_b^j - \boldsymbol{x}_b^i \right\|_2^2 \tag{3}$$

Chaque candidat $\boldsymbol{x}_b^i$ se voit assigner une valeur de qualité entière $Q(\boldsymbol{x}_b^i)$ calculée en utilisant $Inc(\boldsymbol{x}_b^i)$, sa valeur d'inco-



(a) I_{25} (b) I_{40} (c) I_{55} (d) I_{46} (e) *TS* (f) *OG* (g) *TS*+*OG*

FIGURE 3 – Images sources de *MPI S1* [8] (a, b, c) et reconstruction du kiosque de I_{46} (d) à partir de I_{25} avec : (e) traitement statistique (*TS*), (f) optimisation globale (*OG*) résolue par *fusion moves* [9], (g) les deux approches combinées (*TS*+*OG*).

hérence. Celle-ci correspond à la distance *Euclidienne* entre $\boldsymbol{x}_b^i$ et le candidat *inverse* (resp. *direct*) le plus proche si $\boldsymbol{x}_b^i$ est *direct* (resp. *inverse*). Nous souhaitons ainsi assigner une qualité haute aux candidats pour lesquels le vecteur de mouvement correspondant entre I_a et I_b est cohérent avec un vecteur entre I_b et I_a. Le score de qualité $Q(\boldsymbol{x}_b^i)$ correspond à $Inc(\boldsymbol{x}_b^i)$ mis à l'échelle entre 0 et Q_{max} (par rapport aux valeurs d'incohérence maximum et minimum au sein de la distribution) puis arrondi à la valeur entière la plus proche. Lors du calcul des médianes dans l'Eq. (3), chaque candidat $\boldsymbol{x}_b^j$ est considéré $Q(\boldsymbol{x}_b^j)$ fois ce qui se traduit par la prise en compte $Q(\boldsymbol{x}_b^j)$ fois du terme $\left\|\boldsymbol{x}_b^j - \boldsymbol{x}_b^i\right\|_2^2$. Ce mécanisme de vote permet de favoriser les candidats situés dans le voisinage de candidats de bonne qualité et de renforcer la cohérence *forward-backward*.

Le traitement statistique étant appliqué indépendamment pour chaque pixel, nous proposons d'y combiner une méthode d'optimisation globale incluant une régularisation spatiale. L'approche envisagée suit celle décrite dans [10] et permet de sélectionner un champ de mouvement optimal parmi plusieurs champs candidats. Soit $L = \{l_{\boldsymbol{x}_a}\}$ l'étiquettage des pixels $\boldsymbol{x}_a$ de I_a où chaque étiquette définit un candidat de $T_{a,b}(\boldsymbol{x}_a)$. Définissons également $d_{a,b}^{l_{\boldsymbol{x}_a}}$ comme étant les vecteurs de mouvement correspondants aux candidats de $T_{a,b}(\boldsymbol{x}_a)$. La méthode d'optimisation globale consiste à minimiser avec l'algorithme *fusion moves* [9, 10] l'énergie que nous proposons ci-dessous :

$$E_{a,b}(L) = \sum_{\boldsymbol{x}_a} \rho_d(C(\boldsymbol{x}_a, d_{a,b}^{l_{\boldsymbol{x}_a}}(\boldsymbol{x}_a)) + Inc(\boldsymbol{x}_a + d_{a,b}^{l_{\boldsymbol{x}_a}}(\boldsymbol{x}_a))) + \sum_{<\boldsymbol{x}_a, \boldsymbol{y}_a>} \alpha_{\boldsymbol{x}_a \boldsymbol{y}_a} . \rho_r(\left\| d_{a,b}^{l_{\boldsymbol{x}_a}}(\boldsymbol{x}_a) - d_{a,b}^{l_{\boldsymbol{y}_a}}(\boldsymbol{y}_a) \right\|_1) \quad (4)$$

L'énergie considérée inclut un terme de données combinant coût de *matching* $C(\boldsymbol{x}_a, d_{a,b}^{l_{\boldsymbol{x}_a}})$ et valeur d'incohérence $Inc(\boldsymbol{x}_a + d_{a,b}^{l_{\boldsymbol{x}_a}})$ ainsi qu'une régularisation mettant en jeu une similarité de mouvement calculée localement dans I_a (8-connectivité). De plus, α_{x_a, y_a}, ρ_d et ρ_r correspondent respectivement à une similarité de couleur locale dans I_a, à la fonction robuste de *Geman-McClure* et au logarithme négatif d'une distribution *t-Student* [10]. *Fusion moves* fusionne les candidats par paires jusqu'à l'obtention d'un champ optimal $d_{a,b}^*$. Bien qu'incluant une régularisation spatiale, cette méthode ne peut que difficilement être appliquée à un grand ensemble de candidats.

Nous proposons par conséquent de combiner traitement statistique et optimisation globale afin de bénéficier d'une prise en compte de l'information de distribution spatiale, d'une sélection robuste basée sur la qualité intrinsèque des vecteurs de mouvement et d'une régularisation spatiale. En pratique, pour chaque pixel $\boldsymbol{x}_a \in I_a$, nous appliquons le traitement statistique à tout l'ensemble $T_{a,b}(\boldsymbol{x}_a)$. Le critère de minimisation de médiane définie dans l'Eq. (2) permet de sélectionner les N_{opt} meilleurs candidats qui sont ensuite fusionnés par paires par la méthode d'optimisation globale (Eq. (4)) et cela jusqu'à la sélection du meilleur candidat.

TABLE 1 – Comparaison via reconstruction et évaluation PSNR (calcul local pour le kiosque de *MPI S1*, Fig. 3) entre : 1) traitement statistique (*TS*), 2) optimisation globale (*OG*) [9], 3) les deux approches combinées (*TS*+*OG*).

Paires d'images	{25,45}	{25,46}	{25,47}	{25,48}	{25,49}	{25,50}
TS	12.72	15.27	21.7	25.33	**24.48**	24.7
OG	11.19	14	11.14	13.7	21.7	22.22
TS*+*OG	**12.84**	**16.11**	**24.75**	**25.55**	24	**24.79**

4 Résultats

Nos tests concernent des paires d'images $\{I_a, I_b\}$ issues de trois séquences : *MPI S1* [8], *Hope* et *Newspaper*. La construction des candidats a été effectuée via intégration combinatoire *multi-steps* avec en entrée des champs de mouvement élémentaires de *steps* 1, 2, 3, 4, 5, 15 et 30 estimés avec une version 2D de l'estimateur de disparité décrit dans [11].

Une fois l'étape d'intégration combinatoire réalisée, nous avons comparé trois méthodes de sélection : 1) traitement statistique (*TS*), 2) optimisation globale (*OG*) [9], 3) les deux approches combinées (*TS*+*OG*). Les paramètres utilisés sont les suivants : $N_c = 7$, $N_s = 100$, $Q_{max} = 2$, $N_{opt} = 3$. Les champs de mouvement finaux ont d'abord été comparés via reconstruction de I_a à partir de I_b et évaluation PSNR entre l'image reconstruite et I_a pour les régions non-occultées. Tab. 1 et 2 présentent les scores PSNR calculés pour plusieurs paires d'images dans la région du kiosque de *MPI S1* et pour l'image entière dans *Newspaper*. Fig. 3 illustre la reconstruction du kiosque pour une distance de 21 images. Les résultats montrent que *TS* donne de meilleurs résultats que *OG*. La légère amélioration entre *TS*+*OG* et *TS* s'explique par une faible diversité de candidats en sortie du traitement statistique ce qui limite les effets de la régularisation. L'exemple Fig. 3 prouve l'intérêt des chemins de mouvement *multi-steps* qui permettent de « sauter » l'occultation temporaire du kiosque. Les champs de mouvement ont aussi été évalués via compensation en mouvement dans I_b de logos insérés dans I_a (Fig. 4). Dans ce contexte, *TS*+*OG* fonctionne également significativement mieux que *OG*.

(a) Image source I_{5036}

(b) Insertion de logos dans I_{5036}

(c) Propagation à I_{5054}, *OG*

(d) Propagation à I_{5054}, *TS+OG*

(e) Image source I_{230}

(f) Insertion de logos dans I_{230}

(g) Propagation à I_{160}, *OG*

(h) Propagation à I_{160}, *TS+OG*

FIGURE 4 – a-d) Insertion de logos dans I_{5036} et propagation à I_{5054} (*Hope*). e-h) Insertion de logos dans I_{230} et propagation à I_{160} (*Newspaper*). Nous comparons la méthode d'optimisation globale (*OG*) avec le traitement statistique combiné à *OG* (*TS+OG*).

TABLE 2 – Comparaison via reconstruction et évaluation PSNR (calcul sur toute l'image de *Newspaper*, Fig. 4) entre : l'intégration combinatoire suivie de l'optimisation globale (*OG*) [9], du traitement statistique combiné à *OG* (*TS+OG*), la méthode de fusion *multi-steps* (*MSF*) [7].

Paires d'images	{160,190}	{160,200}	{160,210}	{160,220}	{160,230}
OG	21.11	19.33	18.11	17.06	16.29
TS+OG	**21.42**	**19.53**	**18.3**	**17.74**	**17.09**
MSF [7]	20.5	18.22	17.8	16.95	16.6

Enfin, la chaine de traitement complète (intégration combinatoire et sélection *TS+OG*) est plus performante que la méthode de fusion *multi-steps* (*MSF*) [7] comme l'illustrent les scores PSNR Tab. 2. Dans [7], la méthode *MSF* s'est elle-même avérée fournir de meilleurs champs de mouvement que ceux issus de méthodes de l'état de l'art telles que [1, 3, 4].

5 Conclusion

Nous présentons deux contributions pour résoudre le problème de la mise en correspondance dense entre images distantes. Un grand ensemble de champs de mouvement candidats est tout d'abord généré via une méthode d'intégration combinatoire *multi-steps* considérant de multiples chemins de mouvement. Une sélection du meilleur champ de mouvement parmi les candidats générés est ensuite réalisée en combinant traitement statistique et optimisation globale. Dans le contexte de l'édition vidéo, cette nouvelle sélection mène à de meilleurs résultats que des méthodes basées optimisation globale uniquement. De plus, dans son ensemble, l'estimateur proposé permet une meilleure estimation de mouvement dense entre images distantes comparé aux méthodes de l'état de l'art.

Références

[1] C. Zach, T. Pock, and H. Bischof, "A duality based approach for realtime TV-L1 optical flow," *Pattern Recognition*, pp. 214–223, 2007.

[2] D. Sun, S. Roth, and M.J. Black, "Secrets of optical flow estimation and their principles," *IEEE International Conference on Computer Vision Pattern Recognition*, pp. 2432–2439, 2010.

[3] N. Sundaram, T. Brox, and K. Keutzer, "Dense point trajectories by GPU-accelerated large displacement optical flow," *European Conference on Computer Vision*, pp. 438–451, 2010.

[4] T. Brox and J. Malik, "Large displacement optical flow : descriptor matching in variational motion estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 500–513, 2011.

[5] M. W. Tao, J. Bai, P. Kohli, and S. Paris, "SimpleFlow : A non-iterative, sublinear optical flow algorithm," *Computer Graphics Forum*, vol. 31, no. 2, 2012.

[6] T. Crivelli, P.-H. Conze, P. Robert, and P Pérez, "From optical flow to dense long term correspondences," *IEEE International Conference on Image Processing*, 2012.

[7] T. Crivelli, P.-H. Conze, P. Robert, M. Fradet, and P. Pérez, "Multi-step flow fusion : Towards accurate and dense correspondences in long video shots," *British Machine Vision Conference*, 2012.

[8] "Séquence MPI-S1," http ://www.mpi-inf.mpg.de/ granados/projects/vidbginp/index.html.

[9] V. Lempitsky, C. Rother, S. Roth, and A. Blake, "Fusion moves for Markov random field optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010.

[10] V. Lempitsky, S. Roth, and C. Rother, "FusionFlow : Discrete-continuous optimization for optical flow estimation," *IEEE International Conference on Computer Vision Pattern Recognition*, 2008.

[11] P. Robert, C. Thébault, V. Drazic, and P.-H. Conze, "Disparity-compensated view synthesis for s3d content correction," *SPIE IS&T Electronic Imaging Stereoscopic Displays and Applications*, 2012.